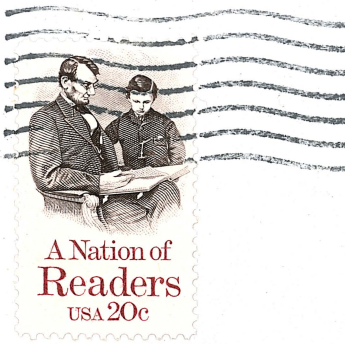
neat notes
THE NEWSLETTER OF NEW ENGLAND APPLE TREE
P.O. BOX 2519
WOBURN, MA. 01888

MIDDLESEX-ESSEX, MA 018
PM
14 JAN
1985



FIRST CLASS MAIL

EXPIRES 03/85

neat notes

THE NEWSLETTER OF NEW ENGLAND APPLE TREE

JANUARY PROGRAM

WPL - Al Scublinsky will continue his discussion of Applewriter, specifically talking about Word Processing Language. WPL can be used to make form letters as well as small adventures.

FEBRUARY PROGRAM: dBase II. Joe Stivaletta will explain the use of this program, as well as providing some examples.

INDEX

Club Information and Disk of the Month ..... Page 2
Opcodes Uttered Here (Anti-Scroll) ..... Page 3
NEAT Happenings and Offers ..... Page 5
Digging Into DOS ..... Page 6
Peeks, Pokes and Calls ..... Page 7
Eamon Adventure Information and Order Form ..... Page 9
Software Library Order Form ..... Page 10

NEAT is the oldest Apple Users group in New England and was founded to provide for the sharing of information between users of home computers. If you own or operate an Apple computer or are thinking about getting involved in this fast growing field, please come to one of our meetings and see what it is all about.

The monthly meetings have several purposes: to conduct club business, to broaden members' knowledge about the possible uses of their Apples, to help newcmers learn about micros, and to provide mutual aid.

Each monthly meeting follows the same format: from 7:00 to 7:30, NEAT volunteers are available to answer questions and offer support. The business meeting follows, lasting until no later than 8:30. Next is the presentation for the evening on a topic of special interest, after which members can pick up their copy of the disk of the month (DOM).

The dues for members are $15 a year and anyone may join at any of the meetings, or by sending an application form to:

New England Apple Tree
P.O. Box 2519
Woburn, MA 01888

Membership in NEAT is a family membership which runs for one year. Members are given a membership card which allows them to attend all meetings and to bring family members and guests. Each membership card also entitles the member to a copy of the DOM on the member's disk at each meeting.

Additional software may be purchase from the NEAT Software Library at member's prices. This software is on disk and can be ordered at the meetings or through the mail and picked up at the meeting or mailed to the member for an additional charge. A copy of NEAT Notes, the club newsletter, is also mailed to all members each month. Membership in NEAT entitles the member to participate in any of the club functions and run for any of the committees or special interest groups. It is in this way that the most benefit is derived from this type of an organization. We all learn from each other and whether it is software, hardware or organizations and people you are interested in, you will gain the most from direct participation.

N.E.A.T. OFFICERS AND CHAIRMEN

President:	Terry Scheller	244-4833
Vice-Pres:	Al Scublinsky	969-1894
Vice-Pres:	Jim Miller	692-7498
Secretary:	Alice Janian	623-6757
Treasurer:	David Lynn	655-6164

PUBLIC DOMAIN LIBRARY

Input and Sales:	G. Janian	623-6757
DOM:	D. Goldstein	273-0567

Membership:	Joe Stivaletta	861-1899

SPECIAL INTEREST GROUPS

SIGED :	Jack Sheehan	924-1197
SIGCP/M :	Joe Stivaletta	861-1899

Editor/Writer: Jim Whitehead	729-0970

Disk of the Month

This month's disk was produced by an Apple group in Dallas, and consists of various Logo programs. These programs all require Apple Logo in order to be run. They may be compatible with other Logo versions, but were not advertised as such. The programs included on this disk are various graphics displays, as well as a hangman program. Pick up this disk at the meeting, or send for it if the meeting passes you by.

Club Information

Diskettes are now on sale for $1.35 per. Disks must be bought in bundles of 25, and will not be broken down. These are Nashua disks, with hub rings and Tyvek sleeves as well as write protect labels. Disks are available at the meetings.

An offer has been made to buy the club's memebersnip list by a company who will then sell that list to companies who will send you Apple product literature. If you are opposed to this, please make your feelings known at the meeting so your name will not be sent.

OPCODES UTTERED HERE
Jim Whitehead

This month's machine language program is called Anti-Scroll. This short routine can be used with your Applesoft programs both while editing them, and during their operation. In short, this routine prevents the contents of the text screen from scrolling up off the top before the user of your programs has a chance to read them. A popular use of a program of this type is exemplified in the Zork series of adventures, where if the message displayed when you enter each room is too long, it will stop after the screen has filled, and ask you for a keypress. After this keypress, the adventure continues.

Anti-Scroll works the same way. Anti-Scroll counts returns, and every 22 returns, it displays the message "More - Hit any key". After this message, it waits for a keypress. Once a key is pressed, Anti-Scroll removes its message, and starts to count returns again. The end effect is that the screen looks just like it would if Anti-Scroll were not present with the added benefit that nothing has been missed because of text scrolling off the top of the screen.

Two main uses of Anti-Scroll are for program debugging, and writing professional looking adventures. While debugging, Anti-Scroll stops program listings from scrolling off the screen. Instead, programmers can page through their programs with ease. Adventure writers can use this routine to make their adventures act like Zork and other Infocom games.

To use this program while debugging, simply type in:

```
] BRUN ANTI-SCROLL
```

from the Applesoft prompt. It will then start working, after initializing itself.

From Applesoft programs, add two lines like this near the start of your program. (Position isn't critical)

```
10 HOME
20 PRINT CHR$(4);"BRUN ANTI-SCROLL"
```

While your program is running, it may be desirable to reset the line count that Anti-Scroll is keeping. The number of returns that Anti-scroll has counted is kept in memory address 10. To reset the line count,

ANTI-SCROLL

```
0300                    ORG $300
0300                    *
0300                    ********************************
0300                    *                              *
0300                    * THIS IS A ROUTINE TO ENSURE  *
0300                    * THAT NO TEXT IS LOST DUE TO  *
0300                    * THE SCROLLING OF THE TEXT    *
0300                    * SCREEN. THIS ROUTINE MAY BE  *
0300                    * USED IN ADVENTURES, PROGRAM  *
0300                    * EDITING, AND OTHER USES WHERE *
0300                    * TEXT SCREEN SCROLLING IS NOT *
0300                    * NEEDED OR SHOULD BE SUPRESSED.*
0300                    * THIS ROUTINE CHECKS SETS UP  *
0300                    * CSW TO POINT AT IT AND THEN  *
0300                    * CHECKS HOW MANY RETURNS ARE  *
0300                    * OUTPUT. IF THE NUMBR SENT IS *
0300                    * THE SAME AS VPOS, THEN THE   *
0300                    * MESSAGE "MORE - HIT ANY KEY" *
0300                    * IS SENT AND THE ROUTINE WAITS*
0300                    * FOR A KEY TO BE PRESSED. IT  *
0300                    * THEN CLEARS THE LINE IT WAS ON*
0300                    * AND RESETS THE NUMBER OF     *
0300                    * RETURNS COUNTED AND LETS OUTPUT*
0300                    * CONTINUE.                    *
0300                    *-----------------------------*
0300                    * NOTE: THIS PROGRAM USES THE  *
0300                    *       SAME LOCATIONS THAT THE *
0300                    *       APPLESOFT USR COMMAND   *
0300                    *       USES AS ITS VECTOR. THIS *
0300                    *       ROUTINE WILL NOT WORK   *
0300                    *       WITH A PROGRAM THAT USES *
0300                    *       THE USR COMMAND!        *
0300                    ********************************
0300                    * BY: JIM WHITEHEAD            *
0300                    * ON: LISA 2.5 ASSEMBLER       *
0300                    * DATE: JANUARY 3, 1984        *
0300                    ********************************
0300                    *
0300                    * === E Q U A T E S ===
0300                    *
03EA                    CSWSETUP EQU $3EA     ;TELLS DOS TO GOTO OUTPUT ROUTINE AT CSW
0016                    VPOS     EQU $22      ;NUMBER OF RETURNSBEFORE MESSAGE IS SENT
002A                    LINE     EPZ $0A      ;RETURN COUNT STORAGE LOC.
0008                    VTEMP    EPZ $08      ;TEMP. STORAGE OF VERTICAL POS.
008D                    RETURN   EQU $8D      ;ASCII RETURN
FF4A                    IOSAVE   EQU $FF4A    ;MONITOR REGISTER SAVE ROUTINE
FF3F                    IOREST   EQU $FF3F    ;BRING BACK FROM ABOVE
F85B                    TABV     EQU $F85B    ;MONITOR TAB ROUTINE
0036                    CSW      EPZ $36      ;OUTPUT SWITCH
FDF0                    COUT1    EQU $FDF0    ;ACTUAL CHAR. OUTPUT ROUTINE
```

simply POKE 10,0. Reseting the line count is desirable, especially when writing adventures. In the following case:

(room message)

] (prompt for a command)

It would be desirable to clear the line count after printing out the room message, so that a keypress is never asked for when it is not necessary, like right after typing in a command line.

In summation, Anti-Scroll is a short routine in machine language that can be used by adventure-writers and programmers alike. The starting address of this program is at $300, decimal 768, and might coincide with other routines. This program cannot be relocated without difficulty, and cannot be used with utilities that make use of the USR command.

ENTERING ANTI-SCROLL

To enter Anti-Scroll, boot up a Dos 3.3 disk, and make your way through the hello program until you are at the Applesoft prompt (]). Then, type the following verbatim, adding a (Return) at the end of each line, and omitting the Applesoft (]) and Monitor (*) prompts.

```
] CALL - 151
* 300:A9 0F 85 36 A9 03 85 37
* 308:A9 00 85 0A 20 EA 03 20
* 310:4A FF A5 45 C9 8D D0 02
* 318:E6 0A A5 0A C9 16 F0 06
* 320:20 3F FF 4C F0 FD A5 25
* 328:85 0B A2 01 BD 6C 03 20
* 330:F0 FD E8 EC 6C 03 F0 02
* 338:D0 F2 A9 00 8D 10 C0 AD
* 340:00 C0 C9 7F B0 02 90 F7
* 348:A9 00 8D 10 C0 A9 00 85
* 350:24 A5 0B 20 5B FB 20 9C
* 358:FC A9 00 85 24 85 0A C6
* 360:0B A5 0B 20 5B FB 20 3F
* 368:FF 4C F0 FD 14 8D 0D 0F
* 370:12 05 A0 AD A0 C8 C9 D4
* 378:A0 C1 CE D9 A0 CB C5 D9
* (ctrl-c)
] BSAVE ANTI-SCROLL,A$300,L$80
```

That's it! Good luck, and happy programming.

ANTI-SCROLL

```
                        *
                        * INIT - INITIALIZE CSW TO POINT
                        *        AT START OF ROUTINE.
                        *
0300                    INIT    LDA #START
0300 A9 0F                      STA CSW
0302 85 36                      LDA /START
0304 A9 03                      STA CSW+1
0306 85 37                      LDA #$00                ;ZERO NUMBER OF LINES.
0308 A9 00                      STA LINE
030A 85 0A                      JSR CSWSETUP
030C 20 EA 03           *
030F                    * START - CHECK FOR LINE 22
030F                    *         IF SO, GOTO WAIT
030F                    *
030F 20 4A FF           START   JSR IOSAVE              ;SAVE ALL
0312 A5 45                      LDA $45                 ;GET A FROM IOSAVE LOCS.
0314 C9 8D                      CMP #RETURN             ;RETURN?
0316 D0 02                      BNE CHECK24             ;YES! ADD A LINE
0318 E6 0A                      INC LINE
031A A5 0A              CHECK24 LDA LINE
031C C9 16                      CMP #VPOS               ;LINE VPOS?
031E F0 06                      BEQ WAIT                ;YES! GOTO WAIT
0320 20 3F FF                   JSR IOREST              ;BRING BACK ALL
0323 4C F0 FD                   JMP COUT1               ;PRINT CHAR.
0326                    *
0326                    * WAIT - PRINT "MORE - HIT ANY KEY"
0326                    *        AND GET KEYPRESS.
0326                    *
0326                    *
0326 A5 25              WAIT    LDA $25                 ;TAKE CUR. SCREEN POS.
0328 85 0B                      STA VTEMP               ;PUT IT IN VTEMP
032A A2 01                      LDX #$01                ;INIT X TO 1 TO PRINT MSSG.
032C BD 6C 03           WAIT1   LDA MESSAGE,X
032F 20 F0 FD                   JSR COUT1
0332 E8                         INX
0333 EC 6C 03                   CPX MESSAGE             ;IS X EQUALL TO MESSAGE LENGTH?
0336 F0 02                      BEQ KEYPRESS            ;YES! GET KEYPRESS
0338 D0 F2                      BNE WAIT1               ;NO GO PRINT MORE
033A A9 00              KEYPRESS LDA #$00               ;CLEAR STROBE
033C 8D 10 C0           K1      STA $C010
033F AD 00 C0           K2      LDA $C000               ;KEYPRESS?
0342 C9 7F                      CMP #$7F                ;KEYPRESS? ()127 = YES!)
0344 B0 02                      BGE K3                  ;YES! GOTO K3
0346 90 F7                      BLT K2                  ;NO! TRY AGAIN
0348 A9 00              K3      LDA #$00                ;CLEAR STROBE
034A 8D 10 C0                   STA $C010
034D A9 00                      LDA #0                  ;SETUP FOR TABV
034F 85 24                      STA $24
```

NEAT SHOW

NEAT will be attending the next Boston Area Microcomputer Show & Fleamarket. Plans include a flea market table and convincing 50 people or more to join NEAT. This show will take place on Saturday, January 19 from 10 Am to 5 Pm. The show is in the Northeast Trade Center, in Woburn. If you would like to donate some of your time for this show, please call Terry Scheller to arrange for a convenient time.

COME ON DOWN!

Article on page 6 taken from Open Apple, a new Apple newsletter.

```
0351- A5 0B     102       LDA VTEMP
0353- 20 5B FB  103       JSR TABV
0356- 20 9C FC  104       JSR !-868       ;CLEAR LINE
0359- A9 00     105       LDA #$00        ;SETUP TO CALL TABV AGAIN
035B- 85 24     106       STA $24
035D- 85 0A     107       STA LINE        ;CLEAR RETURN COUNT
035F- C6 0B     108       DEC VTEMP       ;UP A LINE SO PRINTING WILL NOT BE INTERRUPTED
0361- A5 0B     109       LDA VTEMP
0363- 20 5B FB  110       JSR TABV
0366- 20 3F FF  111       JSR IOREST      ;BRING EM BACK
0369- 4C F0 FD  112       JMP COUT1       ;CONTINUE THE OUTPUT CYCLE
036C- 14 8D     113 MESSAGE HEX 1480
036E- 00 0F 12  114       INV "MORE"
0371- 05                  ASC " - HIT ANY KEY"
0372- A0 AD A0  115
0375- C8 C9 D4
0378- A0 C1 CE
037B- D9 A0 CB
037E- C5 D9
```

Usefull Conversions:

Hex	Decimal
$0300	768
$FF4A	65354
$FB5B	64347
$FDF0	65008
$031D	797
$000A	10

NEAT OFFERS

Being an Apple Club has its ups and downs. One of the ups is that NEAT receives discounts on various products, saving club members' money. Some offers have been pouring in as of late, and tey are as follows. If you have any interest in them, please contact Jim Whitehead or Terry Scheller for more details.

McT SpeeDemon - Makes your Apple II, II+ or IIe run 3.5 times faster, using a 6502C microprocessor. Works in any slot, and is transparent to software. Slows down cards that require it in slots 4, 5, or 6. Special NEAT price is $245.

Rune Software - DOS 4.0, an improved disk operating system for the Apple II, II+ and IIe. DOS 4.0 touts improved disk storage capacity as well as increased loading speed. DOS 4.0 comes with a new plug-in CMOS 6502 prcessor and works with existing DOS 3.3 programs without alteration. For orders of 2 or more, the price is $95.

Topaz press is selling a beginners CP/M book, with a staggered discount. The more who buy, the more the discount. The title of the book is "Strictly for BEGINNERS, CP/M BOOK", and is a CP/M tutorial. Prices are as follows:

2 books - $14.25 each
3 - 5 books - $13.25 each
6 -19 books - $12.25 each
20-49 books - $11.25 each

Plus, the following is a listing of a modem number. This number will connect you with an on-line database of technical employment openings. This board is free to users, and is confidential. No passwords are required. Numbers are as follows:

300 baud	1200 baud
(201) 688-5599	(201) 688-5564
(301) 982-0224	(301) 982-0225

Digging Into DOS

Reading a text file with an Applesoft program has always seemed like a pretty straightforward affair. You *open* the file, tell DOS you want to *read* from it, and *input* it.

If you know how many lines are in the file, you issue that many *input* statements, then stop. Often, however, the number of lines in the file isn't known. In that situation the typical procedure is to read the file until DOS sends an *end of data* message and then stop.

***Onerr goto* errors.** To accomplish this you issue an *onerr goto* statement. When an error occurs, program control will go to your error-handling routine. There you check to see if the error was indeed *end of data* (error number five). If not, you will want to take some true error-handling action. But when the error is simply your signal that the end of the file has been reached, you want to continue processing.

At this point most programmers send control back to their main program. This works for awhile, but if you read enough files in sequence your program will die quite unexpectedly.

For example, try this little program. It tries to read an empty file called *empty* over and over. It reports its efforts as it proceeds. After it tries the 18th time it crashes into the Monitor—fairly unusual for an Applesoft program this simple.

```
100  D$=CHR$(4)
110  N=1
120  F$="EMPTY"

200  PRINT "PROCEEDING WITH READ ";N : N=N+1
500  L=1 : ONERR GOTO 550
510  PRINT D$;"OPEN";F$
520  PRINT D$;"READ";F$

530  INPUT T$(L) : L=L+1 : GOTO 530

550  PRINT : PRINT D$;"CLOSE";F$
560  IF PEEK (222) = 5 THEN GOTO 200

570  PRINT "ERROR #"; PEEK (222);" IN LINE "; PEEK (218) +  PEEK (219) * 256
580  END
```

After the program crashes, return to Applesoft with a *3D0G* and issue a *goto 570* command on your keyboard. Our program will report that we were killed by error number 77, *out of memory*, in line 550.

That may seem pretty incredible to you, given the small size of this program. But people often forget there's more than one kind of memory to run out of. The usual way to run out of memory is to have a program so large and with so many variables that it won't all fit in your computer. For example, just type on your keyboard *dim a(1000,1000)*. An array of that size won't fit and you'll see *?out of memory error* appear on your screen.

All stacked up. The other kind of memory you can run out of is called *stack* memory. Applesoft saves all kinds of stuff on the stack, which is a 256-byte area that the 6502 microprocessor can access quite easily. Whenever you do a *gosub*, for example, Applesoft pushes the current line number on the stack before passing control to the subroutine. That way, when the subroutine's *return* is executed, Applesoft can find its way home by pulling the line number off the stack and jumping to it.

Modify line 500 of our program to *500 gosub 200* and run it again. This will cause a whole bunch of *gosubs* to be executed without any *returns*. You'll again see the *?out of memory error* message; this time during the 26th pass. But unlike when you typed *dim a(1000,1000)*, you've now run out of *stack* memory rather than program memory.

Just as *gosub* pushes a line number on the stack, so does the *onerr goto* command. And the *resume* statement is *onerr goto's return*. The reason our first program crashed was that we never used *resume*.

However, the solution isn't as easy as just beginning to use it. *Resume* sends control back to the line in which the error occurred. If we tried to *resume* after an *end of data* error, we would just get another one and our program would lock up in a tight cycle of *onerr gotos* and *resumes*.

The solution to all this is to use a call to a machine language subroutine within Applesoft that will clear the stack. Use this call in place of *resume*. The call is mentioned in Apple's *Applesoft BASIC Programmer's Reference Manual, Volume 2* on page 265. The subroutine lies at $F328 and can be executed with a *call -3288*.

Earlier Apple manuals included a short machine language program that could be poked into memory and called that would also clear the stack. *Call -3288* does the same thing, and is obviously much easier to execute.

Using this call, you can put the portion of your program that reads files into a subroutine. If you try this without the call, you get a *return without gosub error* when you try to return to your main program after the first *end of data* error.

So here's the standard way to read a text file from Applesoft. This program will run forever or until you press Control-C.

```
100  D$=CHR$(4)
110  N=1
120  F$="EMPTY"
200  PRINT "PROCEEDING WITH READ ";N : N=N+1
210  GOSUB 500
220  GOTO 200

500  L=1 : ONERR GOTO 550
510  PRINT D$;"OPEN";F$
520  PRINT D$;"READ";F$

530  INPUT T$(L) : L=L+1 : GOTO 530

550  PRINT : PRINT D$;"CLOSE";F$
560  IF PEEK (222) = 5 THEN CALL -3288 : RETURN

570  PRINT "ERROR #"; PEEK (222);" IN LINE "; PEEK (218) +  PEEK (219) * 256
580  END
```

Seen In Print

Steve Wozniak told *InfoWorld* to look for the **16-Bit Apple IIx** in 1986. If you missed it, *InfoWorld* carried a report of an interview with Wozniak in its November 19 issue. Wozniak said the machine would have a built-in drive, slots, expandable memory, and the 16-bit 65816 microprocessor. This chip can directly address up to 16 megabytes of memory. Of course it can also emulate a 6502 and thus run today's Apple II software.

In the October issue of *Apple Assembly Line* Bob Sander-Cederlof has a report on the **newly available 65802 microprocessor**. This chip can do internal calculations with 16 bits but handles data eight bits at a time like the 6502. It is completely compatible with the 6502—Bob installed his in a IIe and everything worked normally. Right now the chip costs about $100. The phone number for *Apple Assembly Line* is 214-324-2050.

I've always said ProDOS would become the Apple II operating system of choice when the price of hard disk drives dropped to $500. We're still not there yet, but we're getting closer. A company named **First Class Peripherals** announced a new, $700, 10-megabyte hard disk for the IIe in the November issue of most Apple magazines. The specifications look pretty good, including DOS 3.3 and ProDOS compatibility. We'll be watching this one. Their phone number is 800-538-1307.

The Apple PugetSound Program Library Exchange—the big Apple user group in the Seattle area that publishes *Call-A.P.P.L.E.*—recently changed the legal form of its organization to a cooperative and is now known as **A.P.P.L.E. Co-op**. They've always had really good mail-order prices on hardware and software for Apples; as a co-op they may sell to non-members. Members will get patronage dividends at the end of the year. For people who don't need the hand-holding local dealers supposedly offer, A.P.P.L.E. is a good place to buy stuff. Their phone number is 206-872-2245.

"It must be secret code," thought Holmes, "...it certainly doesn't make sense as a user's manual..."

PEEKS - POKES & CALLS

```
INTEGER BASIC QUICK HIRES CLEAR 10 DIM C$(40):
   20 C$="2000:0 2001(2000.3FFEM E88AG"
   30 FOR I=1 TO LEN (C$)
   40 POKE 511+I,ASC(C$(I))
   50 NEXT I
   60 POKE 72,0
   70 CALL -144
   80 POKE -22482,195
FOR PAGE TWO:
   20 C$="4000:0 4001(4000.5FFEM E88AG"
CLEAR SCREEN:  CALL -936
```

DECIMAL VALUE	ERROR TYPE ENCOUNTERED
0	NEXT WITHOUT FOR
16	SYNTAX
22	RETURN WITHOUT GOSUB
42	OUT OF DATA
53	ILLEGAL QUANTITY
69	OVERFLOW
77	OUT OF MEMORY
90	UNDEFINED STATEMENT
107	BAD SUBSCRIPT
120	REDIMENSIONED ARRAY
133	DIVISION BY ZERO
163	TYPE MISMATCH
176	STRING TOO LONG
191	FORMULA TOO COMPLEX
224	UNDEFINED FUNCTION
254	BAD RESPONSE TO AN INPUT STATEMENT
255	CTRL C INTERRUPT ATTEMPTED

APPLE II PEEKS, POKES, AND CALLS

NOTE: THIS MATERIAL STILL UNDER DEVELOPMENT. "?" INDICATES UNCERTAINTY. COMMENTS AND CORRECTIONS ARE WELCOME.

SCROLLING WINDOW

```
POKE 32,L      SETS LEFT SIDE  OF SCROLLING WINDOW (L=0 TO 39)
POKE 33,W      SETS WIDTH OF SCROLLING WINDOW (W=1 TO 40-L)
POKE 34,T      SET TOP OF SCROLLING WINDOW (T=0 TO 23)
POKE 35,B      SETS BOTTOM OF SCROLLING WINDOW (B=0 TO 23;B)T)
```

CURSOR POSITION

```
POKE 36,CH     SETS HORIZONTAL CURSOR POSITION (CH=0 TO 39)
POKE 37,CV     SETS VERTICAL CURSOR POSITION (CV=0 TO 23)
```

CHARACTER DISPLAY

```
POKE 50,255    WHITE ON BLACK
POKE 50,63     BLACK ON WHITE (INVERSE)
POKE 50,       BLINKING
```

SCREEN FORMAT

GRAPHICS

```
POKE -16304,0  SET COLOR GRAPHICS
POKE -16303,0  SET TEXT MODE
POKE -16302,0  CLEAR MIXED GRAPHICS
POKE -16301,0  SET MIXED GRAPHICS
POKE -16300,0  CLEAR DISPLAY PAGE 2 (BASIC COMMANDS USE
               PAGE 1 ONLY)
POKE -16299,0  SET DISPLAY TO PAGE 2
POKE -16298,0  CLEAR HIRES GRAPHICS
POKE -16297,0  SET HIRES GRAPHIC MODE
CALL -1994     CLEARS UPPER 20 LINES OF PAGE 1 TO REVERSED
               @ SIGNS. IF IN LOW-RES GRAPHICS MODE, THIS
               CLEARS THE UPPER 40 LINES OF GRAPHICS SCREEN
               TO BLACK (PAGE 1 ONLY).
CALL -1998     CLEARS ENTIRE TEXT PAGE 1 TO REVERSED @ SIGNS.
               IF IN PAGE 1 LOW-RES GR, THIS CLEARS THE
               ENTIRE SCREEN TO BLACK.
CALL -12274    CLEARS CURRENT HI-RES SCREEN TO BLACK
CALL -12270    CLEARS CURRENT HI-RES SCREEN TO HCOLOR
               MOST RECENTLY PLOTTED.
               TEXT
CALL -936      HOME CURSOR AND CLEAR SCREEN (ESC-@)
CALL -958      CLEAR FROM CURSOR TO END OF PAGE (ESC-F)
CALL -868      CLEAR FROM CURSOR TO END OF LINE (ESC-E)
CALL -922      LINE FEED (CTRL-J)
CALL -912      SCROLL UP ONE LINE.
```

KEYBOARD

```
X=PEEK (-16384)READ KEYBOARD; IF X(127 THEN KEY WAS PRESSED.
               ALWAYS CLEAR KEYBOARD STROBE AFTER READING
               KEYBOARD.

POKE -16368,0  CLEARS KEYBOARD STROBE.
```

SOUND

```
X=PEEK(-16336) TOGGLES SPEAKER (1 CLICK).
POKE-16336,0
```

CASSETTE

```
X=PEEK(-16352) TOGGLES CASSETTE OUTPUT ONCE (1 CLICK ON
               CASSETTE RECORDING).
```

GAME PADDLES

```
X=PEEK(-16287) READ PDL(0) PUSH BUTTON SWITCH. IF X)127 ,THEN
               SWITCH IS "ON".
X=PEEK(-16286) READ PDL(1) PUSH BUTTON SWITCH. IF X)127 ,THEN
               SWITCH IS "ON".
X=PEEK(-16285) READ PDL(2) PUSH BUTTON SWITCH. IF X)127 ,THEN
               SWITCH IS "ON".

POKE -16296,1  CLEAR GAME I/O AN0 OUTPUT (OFF-3.5V HIGH).
POKE -16295,0  SET GAME I/O AN0 OUTPUT (ON-.3V LOW).
POKE -16294,1  CLEAR GAME I/O AN1 OUTPUT (OFF-3.5V HIGH).
POKE -16293,0  SET GAME I/O AN1 OUTPUT (ON-.3V LOW).
POKE -16292,1  CLEAR GAME I/O AN2 OUTPUT (OFF-3.5V HIGH).
POKE -16291,0  SET GAME I/O AN2 OUTPUT (ON-.3V LOW).
POKE -16290,1  CLEAR GAME I/O AN3 OUTPUT (OFF-3.5V HIGH).
POKE -16289,0  SET GAME I/O AN3 OUTPUT (ON-.3V LOW).
CALL -1250     MONITOR S/R TO READ PADDLE - X-REG CONTAINS
               PADDLE # (0 TO 3)
```

PEEKS, POKES & CALLS

COLORS
LOW-RES

0=BLACK	1=MAGENTA	2=DARK BLUE
3=LIGHT PURPLE	4=DARK GREEN	5=GRAY
6=MEDIUM BLUE	7=LIGHT BLUE	8=BROWN
9=ORANGE	10=GRAY	11=PINK
12=GREEN	13=YELLOW	14=AQUA
15=WHITE		

CALL -2048 PLOT A POINT (AC:Y-COORD Y:X-COORD).
CALL -2023 DRAW A HORIZONTAL LINE.
CALL -2008 DRAW A VERTICAL LINE.
CALL -1998 CLEAR LO-RES SCREEN 1 CLEAR SCREEN-GRAPHICS MODE.
CALL -1994 CLEAR UPPER 20 LINES OF TEXT PAGE 1 TO REVERSE 0. IF IN LOW-RES GR THIS CLEARS UPPER 40 LINES TO BLACK (PAGE 1 ONLY).
CALL -1977 CALCULATE GRAPHICS BASE ADDRESS.
CALL -1953 INCREMENT COLOR BY 2.
CALL -1948 ADJUST COLOR BYTE FOR BOTH HALVES EQUAL.
CALL -1935 MONITOR S/R TO GET SCREEN COLOR (AC:Y-COORD Y:X-COORD). SEE CALL-APPLE DEC 1978.

HI-RES

POKE 800,H SETS HORIZONTAL COORDINATE. H=MODULUS 256
POKE 801,H/256 H=0 (LEFT) TO 279 (RIGHT).
BOTH POKE 800 & 801 ARE REQUIRED.
POKE 802,V SETS VERTICAL COORDINATE.
V=0 (TOP) TO 159 (BOTTOM)
POKE 804,S STARTING ADDRESS OF SHAPE TABLE. S=MODULUS 256
POKE 805,S/256 BOTH 804 & 805 ARE REQUIRED.
POKE 28,C COLOR OF SHAPE.
POKE 806,S SETS SCALE OF SHAPE S(1.
POKE 807,R SETS ROTATION OF SHAPE (R=1 TO 64).
0=RIGHT SIDE UP, 16=90 DEGREES CLOCKWISE.

0=BLACK 1 (0)	1=GREEN (85)	2=VIOLET (170)
3=WHITE 1 (255)	4=BLACK 2	5=ORANGE
6=BLUE	7=WHITE 2	

POKE 812,X SETS COLOR FOR HI-RES.
X=NUMBER OF COLOR IN PARENTHESIS.
CALL 3072 "INIT"-INITIALIZES HI-RES (-12288).
CALL 3986 "CLEAR"-CLEARS THE SCREEN (-12274).
CALL 3780 "PLOT"-PLOTS A POINT ON THE SCREEN (-11580).
CALL?3761 "POSN"-POSITIONS A POINT ON THE SCREEN.
?(-11527) MICRO ?(-11599) PG ?(-11780) SHELDON
CALL?3786 PG "LINE"-DRAWS LINE (X0;Y0;COLOR)
?(-11580) MICRO ?(-11574) PG ?(-11465) PG
CALL -11780 M "FIND" OR ? POSITION
-12272 S "FIND" OR BACKGROUND
(HCOLOR 1 SET FOR BLACK BACKGROUND).
CALL -11471 HI-RES GRAPHICS BKGRND (PARAM=COLOR)

CALL 3805 PG "SHAPE"-DRAWS PREDEFINED SHAPE.
(-11555) PG
CALL -11462 HI-RES DRAW1(X0;Y0;COLOR)
CALL -11335 HI-RES SHLOAD
224-226 HI-RES GR X&Y COORDINATES
228 HI-RES GR COLOR BYTE

OTHER

POKE 72,0?PG GO TO MONITOR
CALL -151 GO TO MONITOR
CALL -3318 CONTINUE
CALL -3774 SAVE
CALL -3776 SAVE INTEGER PROGRAM TO TAPE.
CALL -3973 LOAD INTEGER PROGRAM FROM TAPE.
CALL -4116 RUN INTEGER (KILLS VARIABLES).
CALL -6890 RUN INTEGER (SAVES VARIABLES).
CALL -6729 PLOTS A POINT ON THE SCREEN.
CALL -6739 NEW
CALL -8117 LIST INTEGER PROGRAM.
CALL -8192 RESET INTEGER BASIC. KILLS VARIABLES AND CLEARS WORKSPACE.
CALL -2458 TURN ON MINI-ASSEMBLER.
CALL -2423 SWEET-16 INTERPRETER ENTRY.
CALL -1906 MONITOR S/R DISASSEMBLER ENTRY.
CALL -1728 MONITER S/R-PRINT CONTENTS OF X & Y (REG 9 AS 4 HEX DIGITS.
CALL -1716 MONITOAR S/R PRINT X BLANKS (X REG CONTAINS # TO PRINT).
CALL -1482 MONITOR S/R-IRQ HANDLER.
CALL -1390 MONITOR S/R-BREAK HANDLER.
CALL -1321 MONITOR S/R TO DISPLAY USER REGISTERS.
CALL -1233 MONITOR S/R SCREEN INIT.
CALL -1223 MONITOR S/R SET SCREEN TO TEXT MODE (DESTROYS ACCUMULATOR).
CALL -1216 MONITOR S/R SET GRAPHICS MODE (GR) (DESTROYS ACCUMULATOR).
CALL -1205 MONITOR S/R SET NORMAL WINDOW.
CALL -1181 MONITOR S/R MULTIPLY ROUTINE.
CALL -1148 MONITOR S/R DIVIDE ROUTINE.
CALL -1087 MONITOR S/R CALCULATE TEXT BASE ADDRESS.
CALL -1052 MONITOR S/R SOUND BELL.
CALL -1036 MONITOR S/R MOVE CURSOR RIGHT.
CALL -1027 MONITOR S/R OUTPUT A-REG. AS ASCII ON TEXT SCREEN 1.
CALL -1000 MONITOR S/R TO MOVE CURSOR LEFT.
CALL -998 MONITOR S/R TO MOVE CURSOR UP.
CALL -990 MONITOR S/R PERFORM A VERTICAL TAB TO ROW IN ACCUMULATOR.
CALL -980 MONITOR S/R PERFORM ESCAPE FUNCTION.
CALL -936 MONITOR S/R HOME CURSOR & CLEAR SCREEN (DESTROYS ACCUMULATOR AND Y REGISTER).
CALL -926 MONITOR S/R PERFORM A CARRIAGE RETURN.
CALL -922 MONITOR S/R PERFORM A LINE FEED.
CALL -912 MONITOR S/R SCROLL UP 1 LINE (DESTROYS ACCUM. AND Y REGISTER.
CALL -868 MONITOR S/R CLEAR TO END OF LINE.
CALL -856 WAIT LOOP
CALL -756 GET KEY INPUT FROM KEYBOARD (DESTROYS ACCUMUL. AND Y REGISTER). WAIT FOR KEY PRESS.
CALL -741 MONITOR S/R KEYIN ROUTINE.
CALL -715 READ KEY & PERFORM ESCAPE FUNCTION IF NECESSARY.
CALL -670 PERFORM LINE CANCEL.
CALL -665 PERFORM CARRIAGE RETURN & GET LINE OF TEXT.
CALL -662 GET LINE OF TEXT FROM KEYBOARD (X RETND WITH NUMBER OF CHARACTERS).
CALL -657 WAIT FOR INPUT.

E A M O N A D V E N T U R E L I S T

E01 EAMON MASTER (Version 4)
E02 LAIR OF THE MINOTAUR *
E03 CAVE OF THE MIND *
E04 ZYPHUR RIVER VENTURE *
E05 CASTLE OF DOOM
E06 DEATH STAR *
E07 DEVIL'S TOMB *
E08 ABDUCTOR'S QUARTERS *
E09 ASSAULT ON THE CLONE MASTER *
E10 MAGIC KINGDOM
E11 TOMB OF MOLINAR *
E12 QUEST FOR TREZORE *
E13 CAVES OF TREASURE ISLAND
E14 FURIOSO
E15 HEROES CASTLE *
E16 CAVES OF MONDAMEN
E17 MERLIN'S CASTLE *
E18 HOGARTH CASTLE
E19 DEATH TRAP *
E20 BLACK DEATH
E21 QUEST FOR MARRON
E22 SENATOR'S CHAMBERS
E23 TAMPLE OF NGURCT
E24 BLACK MOUNTAIN *
E25 NUCLEAR NIGHTMARE
E26 ASSAULT ON THE MOLE MAN
E27 REVENGE OF THE MOLE MAN
E28 TOWER OF LONDON
E29 LOST ISLAND OF APPLE
E30 UNDERGROUND CITY
E31 GAUNTLET
E32 HOUSE OF ILL REPUTE * (Obscene)
E33 ORB OF POLARIS
E34 DEATH'S GATEWAY
E35 LAIR OF MUTANTS
E36 CITADEL OF BLOOD
E37 QUEST FOR THE HOLY GRAIL
E38 CITY IN THE CLOUDS
E39 MUSEUM OF UNNATURAL HISTORY
E40 DEAMON'S PLAYGROUND
E41 CAVERNS OF LANST
E42 ALTERNATE BEGINNER'S CAVE
E43 PRIESTS OF XIM
E44 ESCAPE FROM ORC LAIR
E45 SWORD QUEST
E46 LIFE QUEST
E47 FUTURE QUEST
E48 PICNIC IN PARADISE
E49 CASTLE KOPHINOS
E50 THE SEALED DOOR
E51 CAVES OF EAMON BLUFF
E52 DEVIL'S DUNGEON
E53 FEAST OF CARROLL
E54 CRYSTAL MOUNTAIN
E55 MASTER'S DUNGEON
E56 LOST ADVENTURE
E57 MONXOME FOE
E58 LAND OF DEATH
E59 JUNGLES OF VIETNAM
E60 SEWERS OF CHICAGO
E61 HARPY CLOUD
E62 CAVERNS OF DOOM
E63 VALKENBURG CASTLE
E64 MODERN PROBLEMS
E65 SCHOOL OF DEATH
E66 DONGEONS OF XENON
E67 CHAOSIUM CAVES
E68 SMITHS STRONGHOLD

TURNAMENT ADVENTURE

T01 CASTLE OF COUNT FUEY
T02 SEARCH FOR THE KEY-Part I
T03 RESCUE MISSION-Part II

EAMON UTILITIES

ED5 EAMON DESIGNER (Version 5)
EU1 EAMON UTILITY 1
EU2 EAMON UTILITY 2
EU3 EAMON UTILITY 3
EU4 EAMON UTILITY 4

EAMON ADVENTURES

After you have journeyed through the Beginner's Cave on the Eamon System Disk, you will probably wish to continue on in the world of Eamon. Not content to just let your characters collect dust on your floppy disk, you wonder how you can continue the adventure.

Search no further. Available from the club library are a multitude of Eamon adventures designed to play with the character you developed on the system disk. These adventures, listed above are available from NEAT at the regular rates. Simply circle the ones you want, write out a check, wrap up some disks, and you are on your way towards fantastic adventure.

The many places you can journey to include the Lair of the Minotaur, where the rivers and streams are deadly, and a lack of a map can set you going in circles...

Or perhaps the Underground City is more to your liking. Here you will battle electronic guard sentinals and leftovers from Star Trek in an attempt to garner super treasure.

To relive your fondest memories of high school, journey into the School of Death. Here the lunchroom is revealed in its true colors, and the nurse no longer pretends she does not want to hurt you...

For those of you who thought the preist in the Beginner's Cave was a pushover, the Temple of Ngurct is for you. Meet up with a master priest in hand to hand combat. May the best creature win.

The World of Eamon. Another steal from the NEAT library.

NEW ENGLAND APPLE TREE

SOFTWARE LIBRARY ORDER FORM

N.E.A.T. software library disks are available for purchase by members and non-members. The price per one volume copied on N.E.A.T. disk

MEMBERS: $3.00 picked up at the meeting
NON-MEMBERS: $5.00 picked up at the meeting
Add $2.00 per order if mailed to you.

( ) Volume 008 Games
( ) Volume 019 Games
( ) Volume 020 Games
( ) Volume 027 Games
( ) Volume 034 Games
( ) Volume 040 Games
( ) Volume 046 Games
( ) Volume 048 Games
( ) Volume 050 Games
( ) Volume 052 Games
( ) Volume 053 Games
( ) Volume 055 Games
( ) Volume 070 Games
( ) Volume 071 Games
( ) Volume 072 Games
( ) Volume 083 Games
( ) Volume 087 Games
( ) Volume 090 Games
( ) Volume 095 Games
( ) Volume 098 Games
( ) Volume 111 Games
( ) Volume 112 Games
( ) Volume 113 Games
( ) Volume 121 Games
( ) Volume 126 Games
( ) Volume 133 Games
( ) Volume 139 Games
( ) Volume 146 Games
( ) Volume 147 Games
( ) Volume 002 Utilities
( ) Volume 009 Utilities
( ) Volume 036 Utilities
( ) Volume 075 Utilities
( ) Volume 083 Utilities
( ) Volume 086 Utilities
( ) Volume 088 Utilities
( ) Volume 099 Utilities
( ) Volume 100 Utilities
( ) Volume 104 Utilities
( ) Volume 108 Utilities
( ) Volume 116 Utilities
( ) Volume 118 Utilities
( ) Volume 119 Utilities
( ) Volume 127 Utilities
( ) Volume 128 Utilities
( ) Volume 129 Utilities
( ) Volume 131 Utilities
( ) Volume 133 Utilities
( ) Volume 134 Utilities
( ) Volume 159 Utilities
( ) Volume 162 Utilities

( ) Volume 015 Business
( ) Volume 022 Business
( ) Volume 045 Business
( ) Volume 101 Business
( ) Volume 105 Business
( ) Volume 107 Business
( ) Volume 142 Business
( ) Volume 013 Education
( ) Volume 044 Education
( ) Volume 047 Education
( ) Volume 077 Education
( ) Volume 125 Education
( ) Volume 130 Education
( ) Volume 061 Art and Graphics
( ) Volume 084 Art and Graphics
( ) Volume 102 Art and Graphics
( ) Volume 120 Art and Graphics
( ) Volume 132 Art and Graphics
( ) Volume 145 Art and Graphics
( ) Volume 148 Art and Graphics
( ) Volume 152 Art and Graphics
( ) Volume 153 Art and Graphics
( ) Volume 155 Art and Graphics
( ) Volume 156 B. Utility Graphics
( ) Volume 035 Ham Radio
( ) Volume 110 Ham Radio
( ) Volume 136 Ham Radio
( ) Volume 143 Ham Radio
( ) Volume 060 Math
( ) Volume 074 Math
( ) Volume 115 Math
( ) Volume 137 Math
( ) Volume 064 Logo
( ) Volume 065 Logo
( ) Volume 114 Hello Programs
( ) Volume 144 Hello Programs
( ) Volume 063 Modem
( ) Volume 106 Modem
( ) Volume 147 Aviation
( ) Volume 157 Plot Package
( ) Volume 158 Disk Manager
( ) Volume 038 Scientific Disk
( ) Volume 109 Scientific Disk
( ) Volume 140 Food
( ) Volume 081 One-Key DOS
( ) Volume 082 DIVERSI-DOS
( ) Volume 103 Morse Code Tutorial
( ) Volume 141 Astronomy
( ) Volume 023 Utility/Education

( ) Volume 001 Miscellaneous
( ) Volume 003 Miscellaneous
( ) Volume 004 Miscellaneous
( ) Volume 005 Miscellaneous
( ) Volume 006 Miscellaneous
( ) Volume 007 Miscellaneous
( ) Volume 010 Miscellaneous
( ) Volume 011 Miscellaneous
( ) Volume 012 Miscellaneous
( ) Volume 014 Miscellaneous
( ) Volume 016 Miscellaneous
( ) Volume 018 Miscellaneous
( ) Volume 021 Miscellaneous
( ) Volume 023 Miscellaneous
( ) Volume 024 Miscellaneous
( ) Volume 026 Miscellaneous
( ) Volume 028 Miscellaneous
( ) Volume 029 Miscellaneous
( ) Volume 030 Miscellaneous
( ) Volume 031 Miscellaneous
( ) Volume 032 Miscellaneous
( ) Volume 033 Miscellaneous
( ) Volume 039 Miscellaneous
( ) Volume 041 Miscellaneous
( ) Volume 042 Miscellaneous
( ) Volume 043 Miscellaneous
( ) Volume 049 Miscellaneous
( ) Volume 051 Miscellaneous
( ) Volume 054 Miscellaneous
( ) Volume 056 Miscellaneous
( ) Volume 057 Miscellaneous
( ) Volume 059 Miscellaneous
( ) Volume 062 Miscellaneous
( ) Volume 066 Miscellaneous
( ) Volume 067 Miscellaneous
( ) Volume 068 Miscellaneous
( ) Volume 069 Miscellaneous
( ) Volume 073 Miscellaneous
( ) Volume 078 Miscellaneous
( ) Volume 079 Miscellaneous
( ) Volume 080 Miscellaneous
( ) Volume 087 Miscellaneous
( ) Volume 091 Miscellaneous
( ) Volume 092 Miscellaneous
( ) Volume 093 Miscellaneous
( ) Volume 094 Miscellaneous
( ) Volume 096 Miscellaneous
( ) Volume 117 Miscellaneous
( ) Volume 122 Miscellaneous
( ) Volume 124 Miscellaneous
( ) Volume 134 Miscellaneous
( ) Volume 138 Miscellaneous
( ) Volume 150 Miscellaneous
( ) Volume 151 Miscellaneous
( ) Volume 160 Miscellaneous
( ) Volume 161 Miscellaneous
( ) Volume 163 Miscellaneous

CHECK VOLUME DESIRED

NAME .....

ADDRESS .....

CITY, STATE, ZIP .....

Make checks payable to "NEW ENGLAND APPLE TREE"

Send orders to: N.E.A.T.
P.O. BOX 2519
WOBURN, MA. 01888

Membership ( ) RENEWAL $15.00

Membership ( ) NEW.....$15.00

EAMON DISKS ( ) @ $3.00 .....

EAMON DISKS ( ) @ $5.00 .....

LIBRARY DISKS ( ) @ $3.00 .....

LIBRARY DISKS ( ) @ $5.00 .....

TOTAL.....